

200 A

MODUSIM 2.0 USER MANUAL

Part 1. INSTALLATION

1.1 Backup Disk

Before any installation begins, it is always a good practice to backup the program diskette in the back of the report. We assume you are already familiar with DOS commands or Windows operation. For example, in DOS you will need the DISKCOPY command to make backup copies of your program disk.

1.2 System Requirements

To run MODUSIM 2.0, you must have a 386 or 486 based PC with 2MB RAM at least, MS DOS 5.0, Windows 3.0, EXCEL 4.0 and @RISK 3.0. A math co-processor chip is recommended for a 386 based PC, 486 based PCs come with one.

1.3 Installation

To install MODUSIM 2.0, first copy all the files in the attached disk to your hard drive under the directory "c:\MODUSIM". Then you can open the file 'MODU.XLW' directly from EXCEL4.0 & @Risk3.0. Or you can specify the program group name, item name, and the path of MODUSIM to windows. Type WIN to execute Windows, select New from File menu in program Manager to add the program group. The following window will appear, select Program Group and then OK.

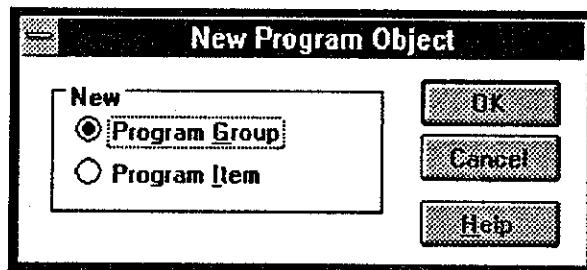


Figure 1.1: Select 'Program Group' for the MODUSIM Program.

Next the following window will appear. Fill in the Description and Group File as indicated. Then select OK.

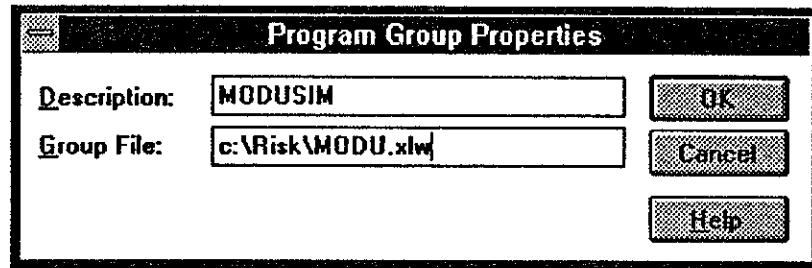


Figure A.1.2: Specify the Group Name and the Filename and Path

Notice that a new program group **MODUSIM** has been created in your Microsoft Windows. Now you can double click the icon to start MODUSIM.

Part 2. USING MODUSIM 2.0

After double clicking the MODUSIM icon, the main window will pop up like the following Figure 2.1. The menu bar is similar to that of EXCEL 4.0, except an extra MODUSIM menu at the end of it. MODUSIM has 18 functions, as in Figure 2.2. Those users who are not familiar with windows operation are recommended to following the step-by-step directions in this chapter. Here, for example, let's say we have a MODU named "Zane Barnes".

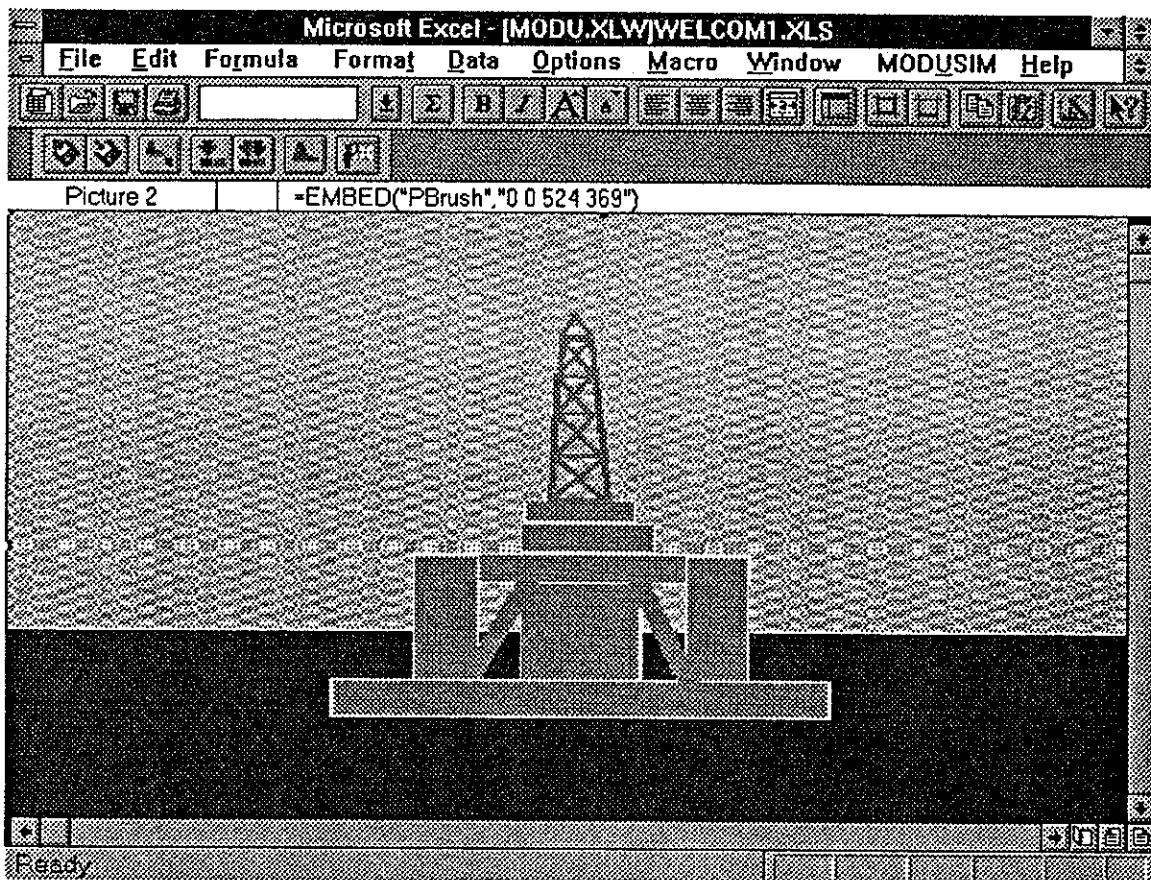


Figure A.2.1 MODUSIM is Popped Up

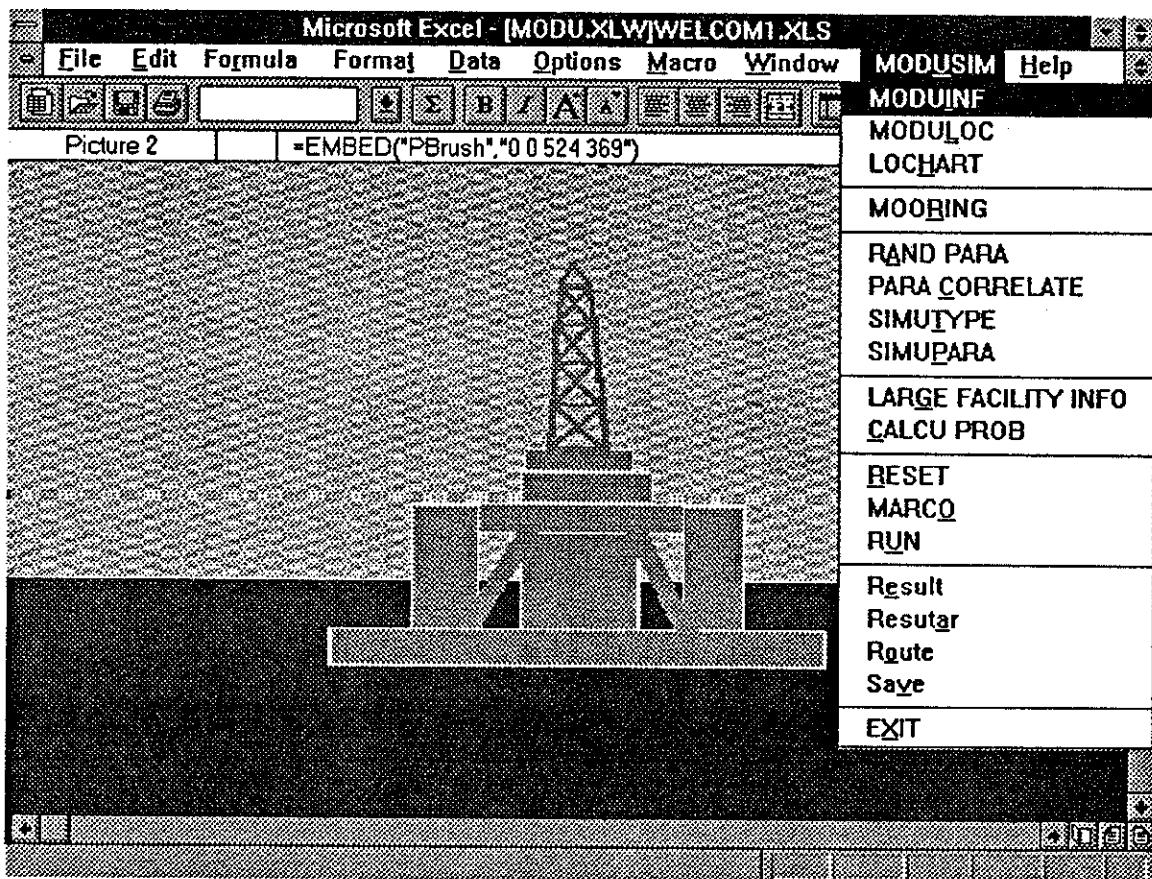


Figure A.2.2 Function in MODUSIM Menu

2.1 MODUINF Command

MODUINF command allows to input the MODU's general information. The dialogue box 'MODU INFORMATION' will pop up when **MODUINF** command is selected.

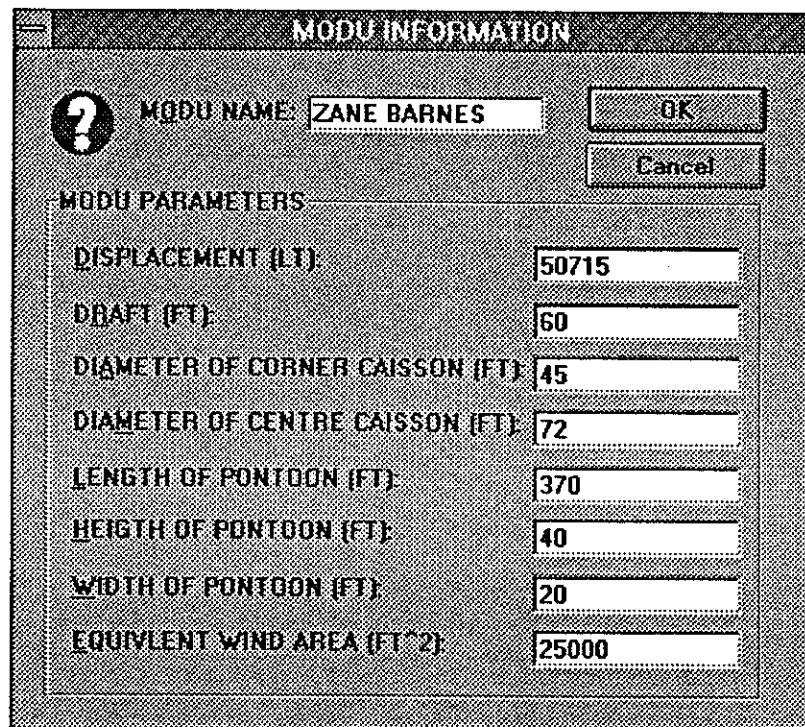


Figure A.2.3 Input MODU General Information

To input the information, you can click on the certain box with mouse or type 'ALT + 'Underline letter'. For example, to input **DISPLACEMENT**, type **ALT+D**. When you finished, click **OK**, or you can click **Cancel** to cancel the dialogue.

2.2 MODULOC Command

MODULOC command allows to input coordinates of MODU's initial location. When **MODULOC** command is selected, the dialogue box 'MODU LOCATION' will pop up.

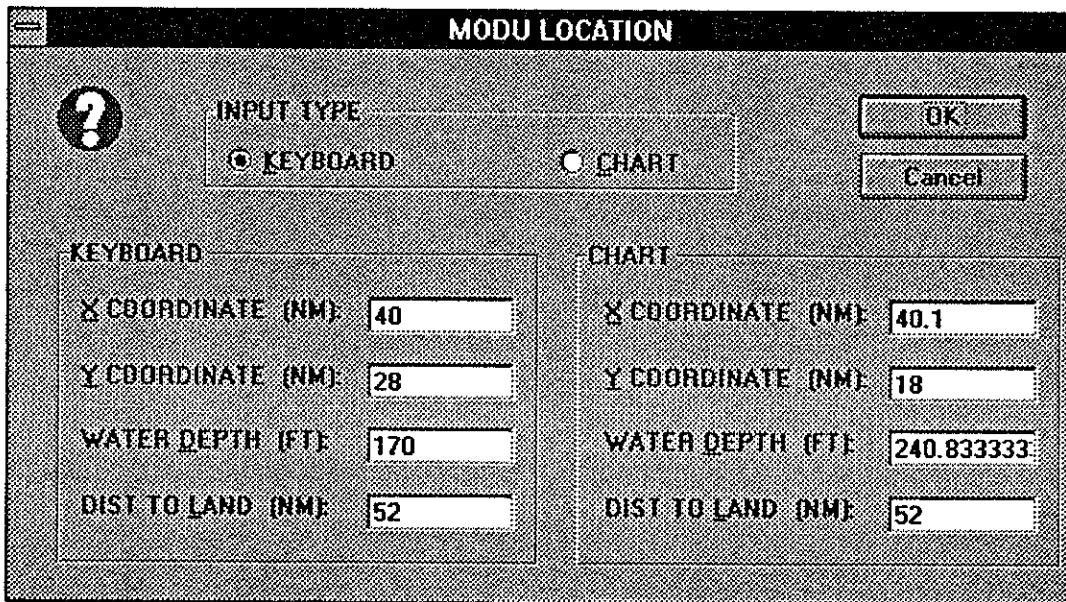


Figure A.2.4 Input MODU Initial Location

In the group of **Input Type**, if **Keyboard** is selected, the information will be input from keyboard to the box in the group of **keyboard**. The input information includes X, Y coordinates, water depth of MODU location and the distance from X-axis to coast. If **Chart** is selected, next command **LOCHART** need to be selected to input information from chart. It is recommended that **Keyboard** function is used to input the initial location and **Chart** function is used to change the location of MODU.

2.3 LOCHART Command

If Chart was selected in the MODULOC command, LOCHART command need to be selected, and the chart 'MODU Moving Route' will pop up. To change the location of the MODU, click on the MODU while hold down CTRL, then drag MODU to wherever you want it to be sited.

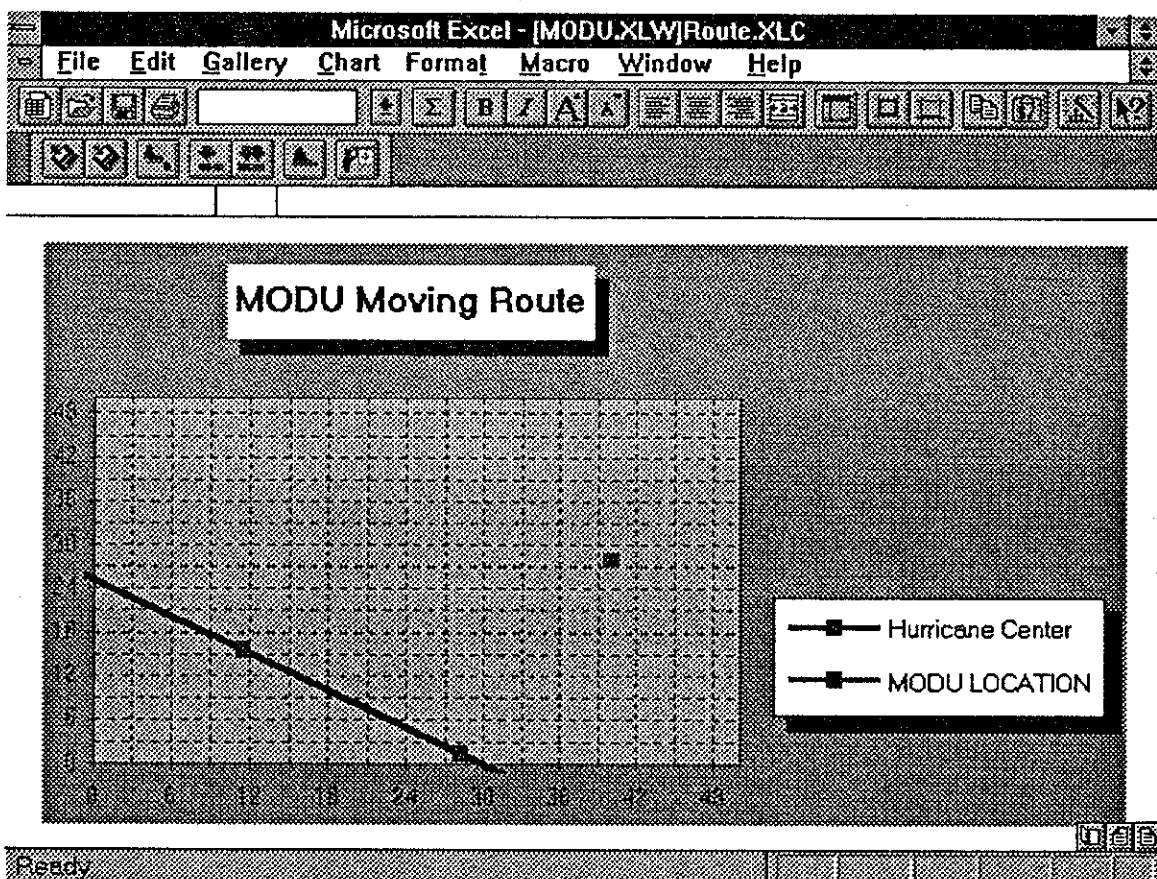


Figure A.2.5 Initial Location of MODU

2.4 Mooring Command

Mooring Command allows to input mooring system information. The dialogue box 'Mooring System' will pop up when Mooring command is selected.

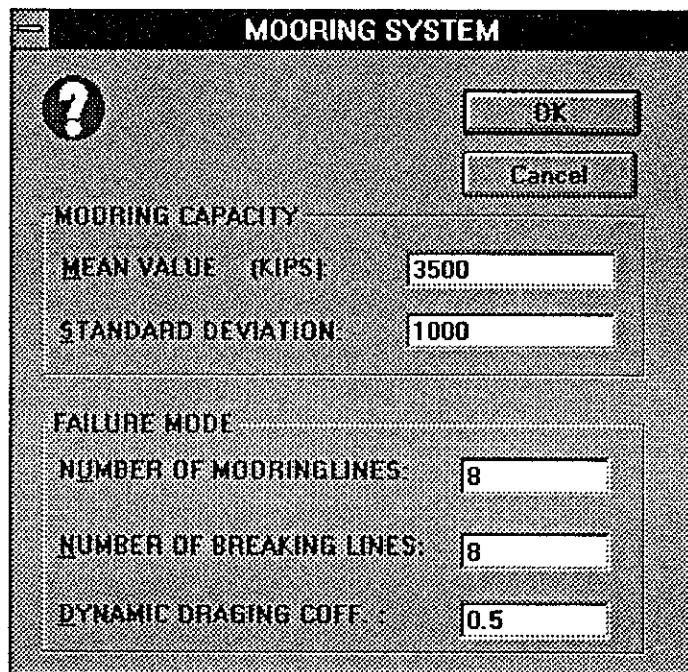


Figure A.2.6 Input Mooring System Information

In the group of **MOORING CAPACITY**, input the mean value and standard deviation of mooring strength; in the group of **FAILURE MODE**, input the total number of mooring lines, number of broken lines while failure and the dynamic dragging coefficient of anchor while they are dragging in the bottom of the sea.

2.5 RAND PARA Command

RAND PARA command allows to input probability distributions of random parameters. The dialogue box 'RANDOM PARAMETER' will pop up when RAND PARA command is selected.

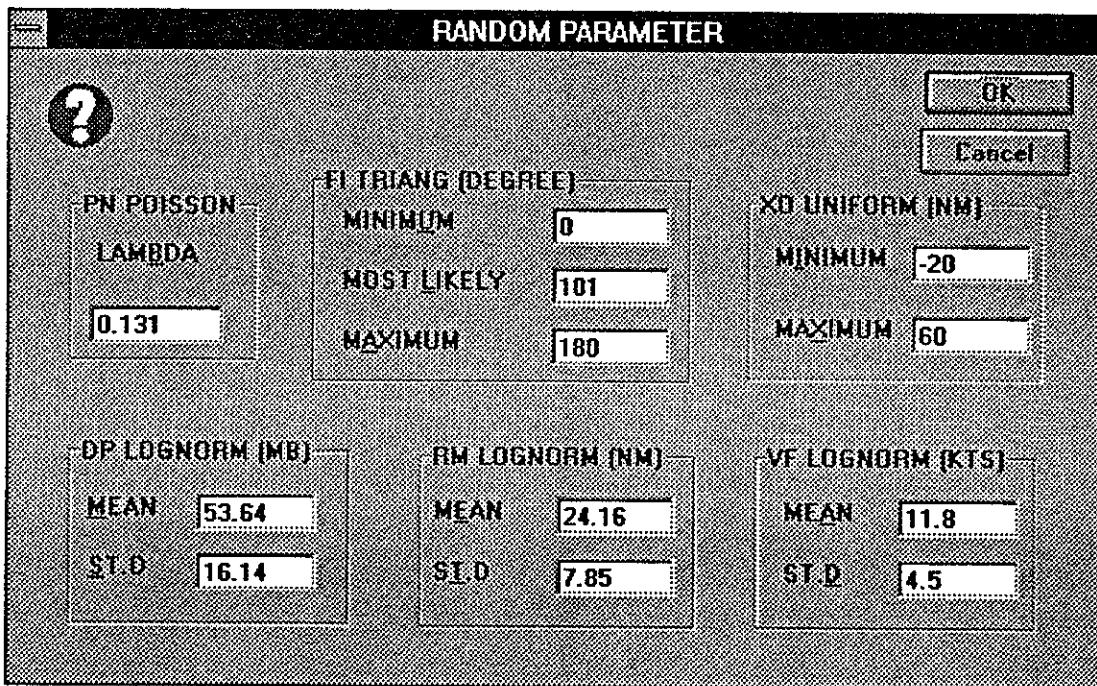


Figure A.2.7 Input Random Parameter Information

Note here, Lamta is the hurricane occurrence rate at a point in the selected reference per year per nautical mile.

2.6 CORR CORRELATE Command

CORR CORRELATE command allows to input correlation among random parameters. The dialogue box 'PARAMETER CORRELATION' will pop up when it is selected.

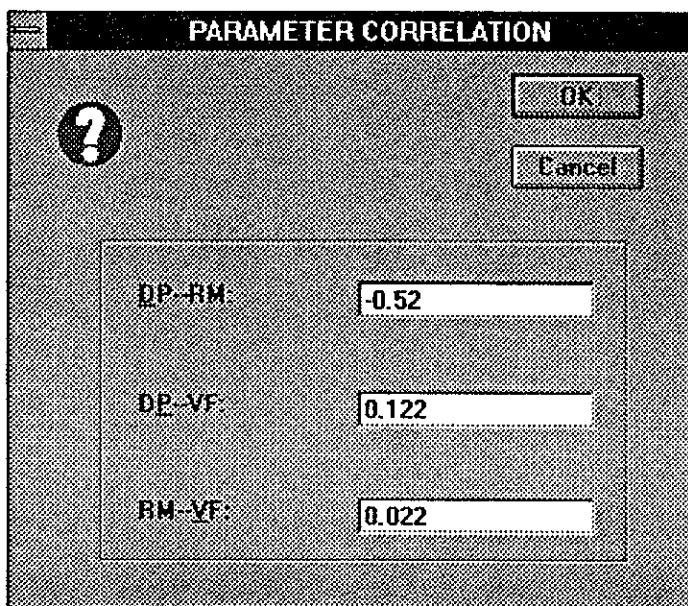


Figure A.2.8 Input Correlation Coefficients among Random Parameters

2.7 SIMUTYPE Command

SIMUTYPE command allows to set up the simulation. The dialogue box 'SIMULATION TYPE' will pop up when SIMUTYPE is selected.

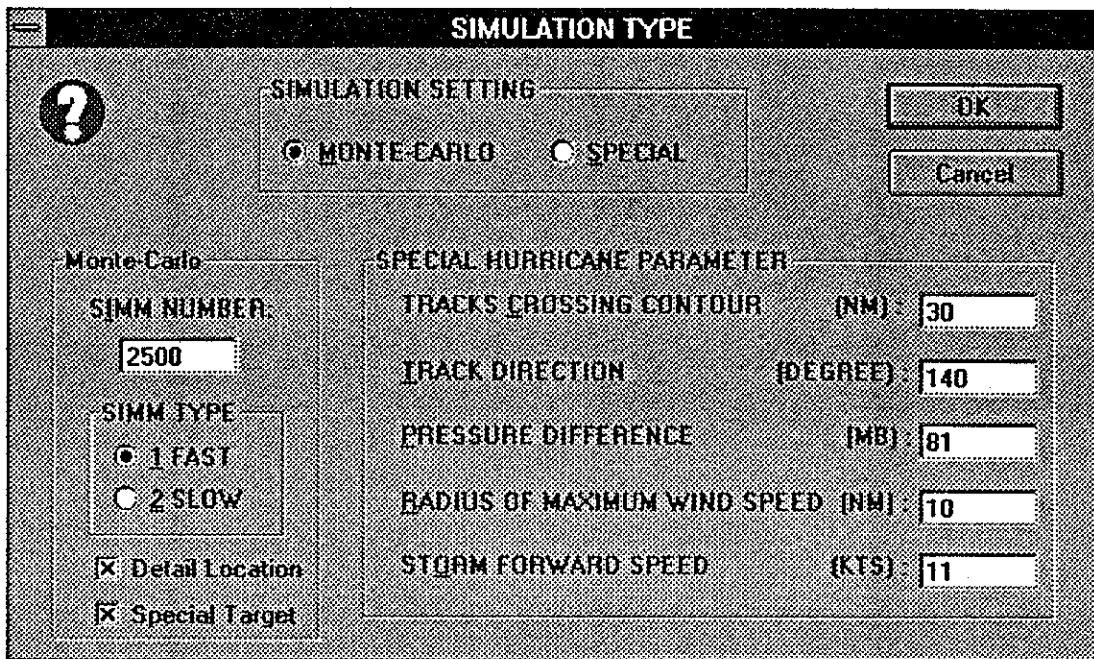


Figure A.2.9 Set Up the Simulation

In the group of **Simulation setting**, select **Monte-Carlo** to calculate the probability of collision, select **SPECIAL** to simulate the MODU's movement during a given hurricane.

In the group of **Monte-Carlo**, input the simulation number, select **1 FAST** in **SIMMTYPE**. The type **2 SLOW** will be discussed later. Check **Detail Location** to include the MODU information within target circles. Check **Special Target** to calculate the probability of collision within a given target.

If **SPECIAL** in **SIMULATION SETTING** was selected, the given hurricane parameters should be input in the group of **SPECIAL HURRICANE PARAMETER**.

2.8 SIMUPARA Command

SIMUPARA command allows to input calculation coefficients. The dialogue box 'SIMULATION PARAMETERS' will pop up when it is selected. Input the wind, wave and current force coefficients in **FORCE PARAMETERS**, Select the type of current velocity distribution in **CURRENT TYPE**, Select the time step between the re-calculation of environmental forces in **TIME STEP**.

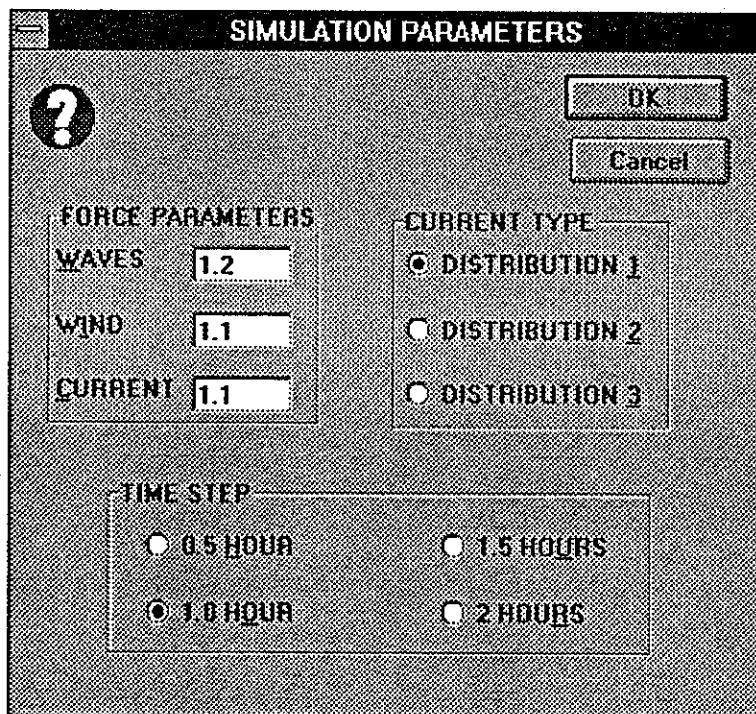


Figure A.2.10 Input Calculation Coefficients

2.9 LARGE FACILITY INFO Command

LARGE FACILITY INFO command allows to set up the simulation for probability of collision within target circles. 'LARGE FACILITY INFORMATION' dialogue box will pop up when it is selected.

Number of Platforms: Structure number within the target circle.

Radius of Circle: Radius of target circle

Radius of safe Distance: The safety distance between the MODU and structures.

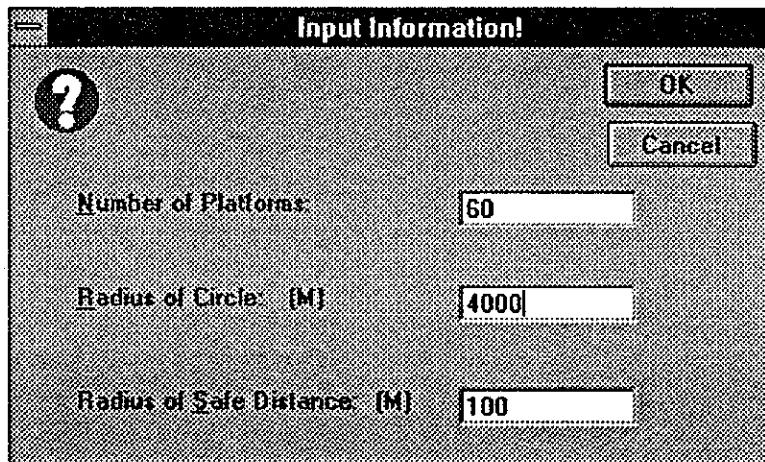


Figure A.2.11 Input Target Circle Information

Note here, the location of target circles is determined by the user from file [MODU.xls]modu.xls. You can choose as many as 5 target circles.

2.10 CALCUL PROB Command

CALCUL PROB command allows to begin the simulation of collision within the target circle. When the simulation is completed, a dialogue box will pop up the calculation result. A pre-calculated curve about the probability of collision within the target circle with R=0.6 to 4.8 nm is presented in Figure ##. For a target circle with given radius and number of structures, the probability of collision can be found from the curve.

2.11 RESET Command

RESET command allows to reset the program before each simulation.

2.12 MARCO Command

MARCO command allows to change active window to marco sheet.

2.13 RUN Command

RUN command is clicked to begin the simulation. Before click RUN, you should set up @RISK simulation parameters. The recommended @RISK simulation settings is as in Figure 2.12. After the simulation is completed, a dialogue box will pop up.

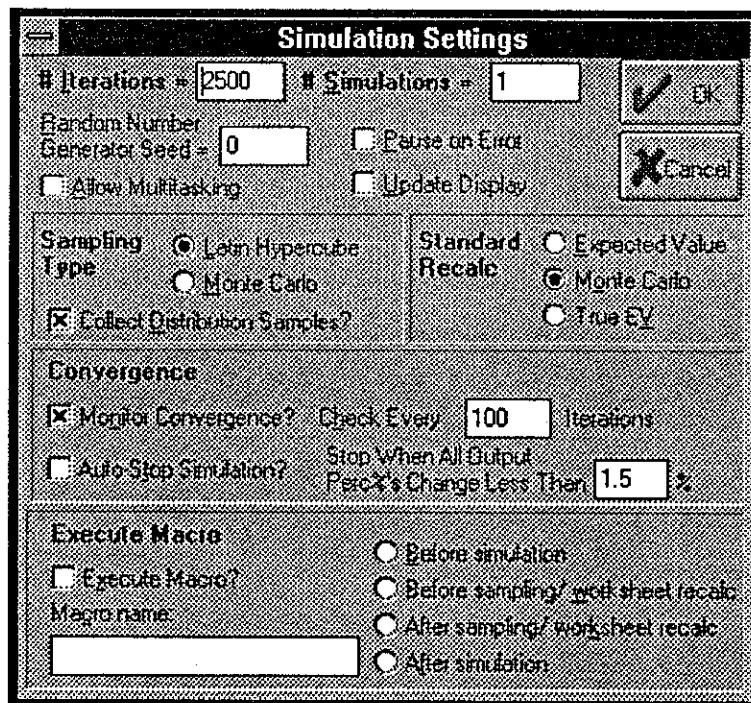


Figure A.2.12 @Risk Simulation Setting

2.14 RESULT Command

Click RESULT command for simulation result.

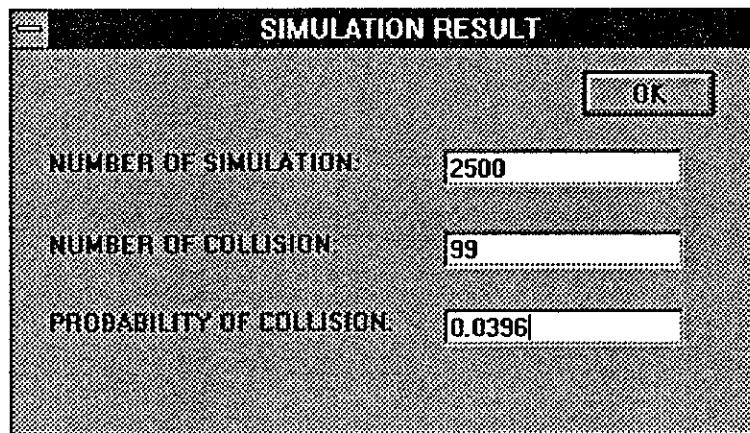


Figure A.2.13 Simulation Result

2.15 RESUTAR Command

Click RESUTAR for output of special target collision probability.

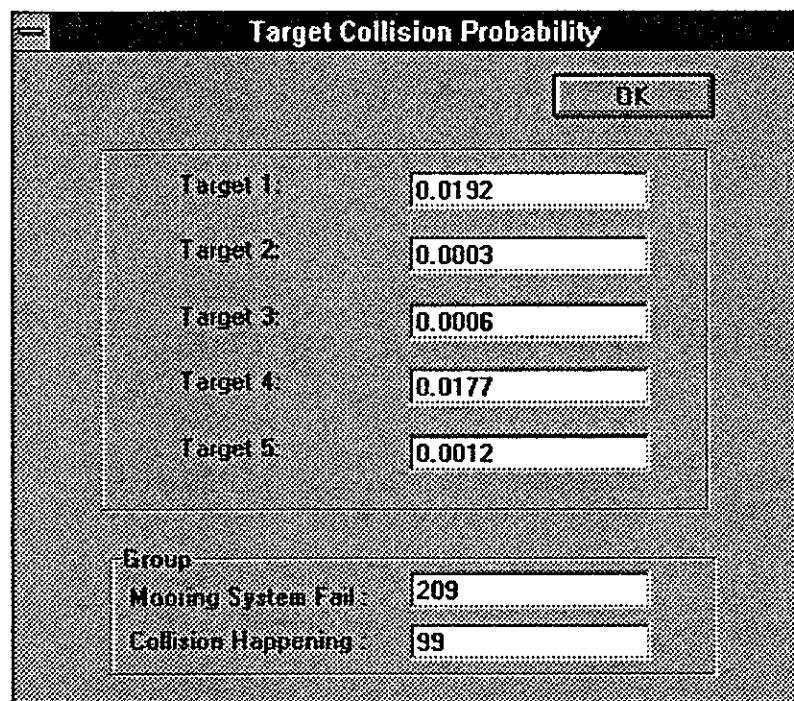


Figure 2.14 Simulation Result for target Circles

2.16 ROUTE Command

In case of simulation the MODU's movement during a given hurricane, click ROUTE command to get the MODU's moving route during hurricanes. Here is Zane Barnes's moving route during hurricane Andrew.

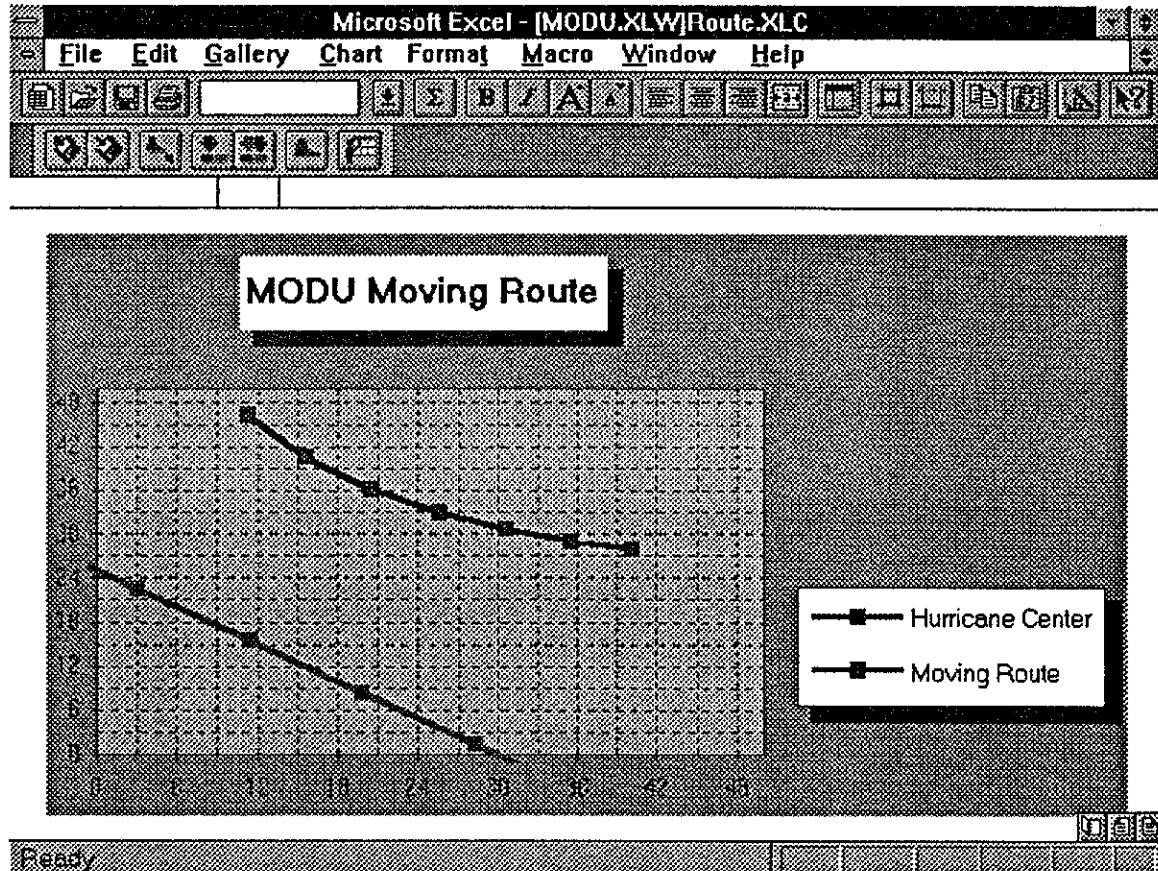


Figure A.2.15 Zane barnes's moving Route during Hurricane Andrew

2.17 SAVE Command

To save simulation result as in Figure 2.16.

SIMULATION RESULT

MODU NAME:		ZANE BARNES			
<hr/>					
LOCATION:		MOORING CAPACITY		FAILURE MODE	
X	Y	MEAN	ST.D	NOM	NOB
50	28.1	3500	1000	8	6
<hr/>					
Probability of collision:			0.0114		
Target 1	Target 2	Target 3	Target 4	Target 5	Mooring
0.0048	0.0006	0.0048	0.0006	0	0.09
<hr/>					

Figure A.2.16 saved simulation Result

2.18 EXIT Command

Click Exit to close the application.

MODUSIM MARCO CODE

Auto-start Command

```
START()
=ADD.MENU(1,15;M28,9)
=WORKBOOK.ACTIVATE("[MODU.XLW]WELCOM1.XLS")
=RETURN()
```

Input MODU General Information

```
MDINFO()
=DIALOG.BOX(MODUINF)
=RETURN()
```

Input MODU Location

```
MDLOC()
=FORMULA("=(KDTL-CYMI)*KWDMI/(KDTL-KYMI)",MODUSIM.XLMIJ58)
=FORMULA("=KDTL",MODUSIM.XLMIJ60)

=DIALOG.BOX(MODULOC)
=ACTIVATE("[MODU.XLW]MODUSIM.XLM")
=FORMULA("=(KDTL-CYMI)*KWDMI/(KDTL-KYMI)",IJ58)
=FORMULA("=KDTL",IJ60)

=RETURN()
```

Input Simulation Pre-setting

```
SIMTYPE()
=DIALOG.BOX(SIMUTYPE)
=RETURN()
```

Input Simulation Parameter

```
SIMPARA()
=DIALOG.BOX(SIMUPARA)
=RETURN()
```

Input Random parameter

```
RANDPRO()
=DIALOG.BOX(RANDPA)
=RETURN()
```

Input Correlation between Random Parameter

```
PRCORR()  
=DIALOG.BOX(PACORR)  
=IF(ALERT("CHANGE CORRELATION PARAMETERS?",1))  
= Risk.Correlate(MODUSIM.XLMIB297,1,DPRM,MODUSIM.XLMIB298,1)  
= Risk.Correlate(MODUSIM.XLMIB297,1,DPVF,MODUSIM.XLMIB299,1)  
= Risk.Correlate(MODUSIM.XLMIB298,1,RMVF,MODUSIM.XLMIB299,1)  
=END.IF()  
=RETURN()
```

Input Mooring Capacity

```
MOOR()  
=DIALOG.BOX(MOORING)  
=RETURN()
```

Stop by User when in Special Simulation Case

```
STOPP()  
=IF(ALERT("STOP SIMULATION?",1))  
= HALTO  
= WORKBOOK.ACTIVATE("[MODU.XLW]WELCOM1.XLS")  
=END.IF()  
=RETURN()
```

Reset the Program

```
MRESET()  
  
=SET.NAME("COLLISIONa",0)  
=SET.NAME("COLLISION",0)  
=SET.NAME("SIMNUM",0)  
=IF(INTYPE=1)  
= SET.NAME("XMI",KXMI)  
= SET.NAME("YMI",KYMI)  
= SET.NAME("WDMI",KWDMI)  
= SET.NAME("DTL",KDTL)  
=ELSE()  
= SET.NAME("XMI",CXMI)  
= SET.NAME("YMI",CYMI)  
= SET.NAME("WDMI",CWDMI)  
= SET.NAME("DTL",CDTL)  
=END.IF()  
=SET.NAME("XWD", (DTL*WDMI)/(DTL-YMI))  
=WORKBOOK.ACTIVATE("[MODU.XLW]WELCOM1.XLS")  
=RETURN()
```

Change Active Window to Marco Sheet

```
MARCO()
=FORMULA.GOTO(MODUSIM.XLMIA1)
=RETURN()
```

Begin the Simulation

```
RUN()
=SET.NAME("COLLISIONa",0)
=SET.NAME("COLLISION",0)
=SET.NAME("SIMNUM",0)
=SET.NAME("Tarcoll1",0)
=SET.NAME("Tarcoll2",0)
=SET.NAME("Tarcoll3",0)
=SET.NAME("Tarcoll4",0)
=SET.NAME("Tarcoll5",0)
=SET.NAME("moorfail",0)
=IF(INTYPE=1)
=  SET.NAME("XMI",KXMI)
=  SET.NAME("YMI",KYMI)
=  SET.NAME("WDMI",KWDMI)
=  SET.NAME("DTL",KDTL)
=ELSE()
=  SET.NAME("XMI",CXMI)
=  SET.NAME("YMI",CYMI)
=  SET.NAME("WDMI",CWDWI)
=  SET.NAME("DTL",CDTL)
=END.IF()
=SET.NAME("XWD", (DTL*WDMI)/(DTL-YMI))

=IF(AND(ST=1,STT=2))
=  SET.NAME("KKKK",1)
=  Risk.Add.Output('B:\[MODU.XLW]MODU.XLS'!$C$3:$D$6)
=  Risk.Simulate(1,50,2,1,0,2,0,0,2,"[modu.xls]modusim.xls!simulate",0,0)
=ELSE.IF(AND(ST=1,STT=1))
=  SET.NAME("KKKK",SIMM)
=  SIMULATE()
=  ALERT("SIMULATION FINISHED",3)
=ELSE()
=  ACTIVATE("[MODU.XLW]MODU.XLS")
=  SELECT("R3C3:R40C6")
=  CLEAR(3)
=  WORKBOOK.ACTIVATE("[modu.xls]Route.xls")
=  SET.NAME("KKKK",1)
=  SIMULATE()
=  DIALOG.BOX(okk)
=  WORKBOOK.ACTIVATE("[MODU.XLW]WELCOM1.XLS")
```

```
=END.IF()
=RETURN()
```

OK to Return
OK() = FORMULA.GOTO('B:\[MODU.XLW]WELCOM1.XLS'!A1) =RETURN()

Simulation Output (Probability of Collision)
Result() =IF(simnum=0) =SET.NAME("SIMNUM",1) =END.IF() =IF(ST=1) = FORMULA("=SIMNUM",MODUSIM.XLMIJ172) =IF(Del) = FORMULA("=COLLISION",MODUSIM.XLMIJ174) = FORMULA("=COLLISION/SIMNUM",MODUSIM.XLMIJ176) =ELSE() = FORMULA("=COLLISIONa",MODUSIM.XLMIJ174) = FORMULA("=COLLISIONa/SIMNUM",MODUSIM.XLMIJ176) =END.IF() = DIALOG.BOX(DRESULT) =ELSE() = Route() =END.IF() =RETURN()

Simulation Output (Target Collision Probability)
Resutar() = FORMULA("=Tarcoll1/SIMNUM",MODUSIM.XLMIJ242) = FORMULA("=Tarcoll2/SIMNUM",MODUSIM.XLMIJ244) = FORMULA("=Tarcoll3/SIMNUM",MODUSIM.XLMIJ246) = FORMULA("=Tarcoll4/SIMNUM",MODUSIM.XLMIJ248) = FORMULA("=Tarcoll5/SIMNUM",MODUSIM.XLMIJ250) = FORMULA("=moorfail",MODUSIM.XLMIJ254) = FORMULA("=collision",MODUSIM.XLMIJ256) =DIALOG.BOX(Tarprop) =RETURN()

Draw the MODU Moving Route

Route()

```
=WORKBOOK.ACTIVATE("[modu.xls]Route.xls")  
  
=EDIT.SERIES(1,"Moving Route",'B:\[MODU.XLW]MODU.XLS'!$C$3:$C$40,  
'C:\MODUSIM\[MODU.XLW]MODU.XLS'!$D$3:$D$40,,1)  
=EDIT.SERIES(2,"Hurricane Center",'B:\[MODU.XLW]MODU.XLS'!$E$3:$E$40,  
'C:\MODUSIM\[MODU.XLW]MODU.XLS'!$F$3:$F$40,,1)  
=DIALOG.BOX(okk)  
= WORKBOOK.ACTIVATE("[MODU.XLW]WELCOM1.XLS")  
=RETURN()
```

Save the Simulation Result

SAVE()

```
=FORMULA("=[modu.xls]modusim.xls!moduname",'B:\[MODU.XLW]RESULT.XLS'!D6)  
  
=IF(INTYPE=1)  
= FORMULA("=[modu.xls]modusim.xls!KXMI",'B:\[MODU.XLW]RESULT.XLS'!B10)  
= FORMULA("=[modu.xls]modusim.xls!KYMI",'B:\[MODU.XLW]RESULT.XLS'!C10)  
=ELSE()  
= FORMULA("=[modu.xls]modusim.xls!CXMI",'B:\[MODU.XLW]RESULT.XLS'!B10)  
= FORMULA("=[modu.xls]modusim.xls!CYMI",'B:\[MODU.XLW]RESULT.XLS'!C10)  
=END.IF()  
  
=FORMULA("=[modu.xls]modusim.xls!IMMV",'B:\[MODU.XLW]RESULT.XLS'!D10)  
=FORMULA("=[modu.xls]modusim.xls!MSTD",'B:\[MODU.XLW]RESULT.XLS'!E10)  
=FORMULA("=[modu.xls]modusim.xls!NOM",'B:\[MODU.XLW]RESULT.XLS'!F10)  
=FORMULA("=[modu.xls]modusim.xls!NOB",'B:\[MODU.XLW]RESULT.XLS'!G10)  
=FORMULA("=[modu.xls]modusim.xls!Prob",'B:\[MODU.XLW]RESULT.XLS'!E12)  
  
=FORMULA("=[modu.xls]modusim.xls!Tar1",'B:\[MODU.XLW]RESULT.XLS'!B14)  
=FORMULA("=[modu.xls]modusim.xls!Tar2",'B:\[MODU.XLW]RESULT.XLS'!C14)  
=FORMULA("=[modu.xls]modusim.xls!Tar3",'B:\[MODU.XLW]RESULT.XLS'!D14)  
=FORMULA("=[modu.xls]modusim.xls!Tar4",'B:\[MODU.XLW]RESULT.XLS'!E14)  
=FORMULA("=[modu.xls]modusim.xls!Tar5",'B:\[MODU.XLW]RESULT.XLS'!F14)  
  
=FORMULA("=[modu.xls]modusim.xls!mofail/[modu.xls]modusim.xls!kkkk",)  
'C:\MODUSIM\[MODU.XLW]RESULT.XLS'!G14)  
=WORKBOOK.COPY("[modu.xls]result.xls","result.xls",1)  
=WORKBOOK.ACTIVATE("[modu.xls]result.xls")  
  
=RETURN()
```

Input the MODU Initial Location from Chart

```
LOCHART
=WORKBOOK.ACTIVATE("[modu.xls]Route.xls")
=IF(INTYPE=1)
= EDIT.SERIES(2,"MODU LOCATION",MODUSIM.XLM!$J$51:$J$51,MODUSIM.XLM!#REF!,,2)
= DIALOG.BOX(okk)
= WORKBOOK.ACTIVATE("[MODU.XLW]WELCOM1.XLS")

=ELSE.IF(INTYPE=2)
= EDIT.SERIES(2,"MODU LOCATION",MODUSIM.XLM!$J$54:$J$54,MODUSIM.XLM!$J$56:$J$56,,2)
=END.IF()
=RETURN()
```

Quit the Program

```
EXIT
=DELETE.MENU(1,9)
=CLOSE.ALL()
=RETURN()
```

Main Simulation Code

SIMULATE

```
=SET.NAME("DT",DTT*0.5)

=FOR("SS",1,KKKK)
=SET.NAME("mor",0)

=IF(ST=1)
= SET.NAME("PN",RiskPoisson(PNLAM))
=ELSE()
= SET.NAME("PN",1)
=END.IF()

=FOR("N",1,PN)

=IF(OR(STT=2,ST=2))
=ACTIVATE("[MODU.XLW]MODU.XLS")
=SELECT("R3C3:R40C4")
=CLEAR(3)
=END.IF()

=SET.NAME("COLL",0)
=SET.NAME("STOP",0)
=SET.NAME("XM",XMI)
=SET.NAME("YM",YMI)
```

```

=SET.VALUE(REPORT,1)
=SET.VALUE(CONT,1)

=IF(ST=1)
= SET.NAME("XO",RiskUniform(XOMIN,XOMAX))
= SET.NAME("FI",RiskTriang(FIMIN,FILIKE,FIMAX)*PI()/180)
= SET.NAME("DP",RiskLognorm(DPMEAN,DPSTD))
= SET.NAME("RM",RiskLognorm(RMMEAN,RMSTD))
= SET.NAME("VF",RiskLognorm(VFMEAN,VFSTD))
= SET.NAME("MMV",RiskLognorm(IMMV,MSTD)*1000)

=ELSE()
= SET.NAME("XO",Sxo)
= SET.NAME("FI",SFI*PI()/180)
= SET.NAME("DP",SDP)
= SET.NAME("RM",SRM)
= SET.NAME("VF",SVF)
= SET.NAME("MMV",IMMV*1000)
=END.IF()

=IF(ABS(FI-PI()/2)<0.001)
= IF(ABS(XM-XO)>(10*RM))
= SET.NAME("XH",1000)
= ELSE()
= SET.NAME("XH",XO)
= SET.NAME("YH",YM-SQRT((10*RM)^2-(XM-XO)^2))
= END.IF()
=ELSE.IF(FI<>(PI()/2))
= SET.NAME("XH",FUNXH(XM,YM,FI,XO,RM))
= SET.NAME("YH",TAN(FI)*(XH-XO))
=END.IF()

=IF(XH=1000,GOTO(NEXT))

=SET.NAME("KK",0)
=SUBTOTFOR()
=IF(OR(YH>DTL,YM>DTL,NDR>11,DR>(WDM+SUR+HM/2)),GOTO(NEXT))

=WHILE((TOTALF-MMV)<0)

=SET.NAME("XH",XH+VF*DT*COS(FI))
=SET.NAME("YH",YH+VF*DT*SIN(FI))
=IF(OR(YH>DTL,YM>DTL,NDR>11,DR>(WDM+SUR+HM/2)),GOTO(NEXT))

=SET.NAME("KK",0)
=SUBTOTFOR()

=IF(OR(YH>DTL,YM>DTL,NDR>11,DR>(WDM+SUR+HM/2)),GOTO(NEXT))

```

```

=NEXT()

=IF(mor=0)
=SET.NAME("moorfail",moorfail+1)
=SET.NAME("mor",1)
=END.IF()

=FOR("NN",1,10000)

=IF(OR(ST=2,STT=2))
=SUBWRITE()
=END.IF()

=IF(VM<>0)
=SUBCOLLISION()
=ELSE()
=SET.NAME("COLL",0)
=END.IF()

=IF(AND(ST=1,COLL=1),GOTO(nextt))
=IF(AND(ST=2,STOP=1),GOTO(nextt))
=IF(OR(YH>DTL,YM>DTL,NDR>11,DR>(WDM+SUR+HM/2)),GOTO(NEXT))

=SET.NAME("XH",XH+VF*DT*COS(FI))
=SET.NAME("YH",YH+VF*DT*SIN(FI))

=SET.NAME("XM",XM+VM*DT*COS(WD))
=SET.NAME("YM",YM+VM*DT*SIN(WD))

=SET.NAME("KK",1)
=SUBTOTFOR()

=NEXT()

=NEXT()

=SET.NAME("simnum",simnum+1)

=IF(OR(ST=2,STT=2))
=ACTIVATE("[MODU.XLW]MODU.XLS")
=SELECT("r3c3:r40c6")
=COPY()
=PASTE.SPECIAL(3,1,FALSE,FALSE)
=END.IF()

=NEXT()

=RETURN()

```

Environment Force Calculation

SUBTOTFOR()

```
=SET.NAME("CTA",FUNCTA(XM,YM,XH,YH,FI))
=SET.NAME("NDR",FUNNDR(XM,YM,XH,YH,RM))

=IF(NDR>11,RETURN())

=SET.NAME("W",FUNWINSPEE(DP,RM,VF,CTA,NDR))
=SET.NAME("WD",FUNWINDIR(CTA))
=SET.NAME("WDM",FUNWDM(WDMI,YMI,YM,DTL))
=SET.NAME("CV",FUNCURVEL(W,WDM))
=SET.NAME("HM",FUNWAVHEI(W,WDM))
=SET.NAME("T",FUNPERIOD(W,CTA))
=SET.NAME("SUR",FUNSURGE(HM,WDM))

=SET.NAME("CUF",FUNCURFOR(PARACUR,DR,SUR,CV,DCC,DCE,LPO,HPO,CUD))
=SET.NAME("WINF",FUNWINFOR(PARAWIN,W,EWA))
=SET.NAME("WAVDF",FUNWAVDRI(HM))

=SET.NAME("WAF",FUNWAVFOR(PARAWAV,DR,SUR,WM,T,WDM,DCC,DCE,LPO,HPO,WPO))
=SET.NAME("TOTALF",WAF+WINF+CUF+WAVDF)

=IF(KK=1,GOTO(KKK))
=IF(TOTALF<MMV)
= RETURN()
=END.IF()

=IF(DR>(WDM+SUR))

= IF(NOM=NOB)
=   SET.NAME("ACCE",TOTALF*9.8/2.2/(1000*DISP))
=   SET.NAME("VM",0.5*ACCE*T/4*1.944)
= ELSE.IF(NOM<>NOB)
=   SET.NAME("DRAGF",2*DMP*MMV*(NOM-NOB)/NOM)
=   IF(TOTALF<DRAGF)
=     SET.NAME("VM",0)
=   ELSE.IF(TOTALF>DRAGF)
=     SET.NAME("ACCE",TOTALF-DRAGF)*9.8/2.2/(1000*DISP))
=     SET.NAME("VM",0.5*ACCE*T/4*1.944)
=   END.IF()
= END.IF()

=ELSE.IF(DR<(WDM+SUR))

= IF(NOM=NOB)
```

```

= IF(CUD=1)
=   SET.NAME("VM",SQRT((WINF+WAVDF)/(2.85*((4*DCC+DCE)*(DR+SUR)*0.7+2*LPO*HPO)))+
+CV)
= ELSE.IF(CUD=2)
=   SET.NAME("VM",SQRT((WINF+WAVDF)/(2.85*((4*DCC+DCE)*(DR+SUR)*0.7+2*LPO*HPO)))+
+CV*0.75)
= ELSE.IF(CUD=3)
=   SET.NAME("VM",SQRT((WINF+WAVDF)/(2.85*((4*DCC+DCE)*(DR+SUR)*0.7+2*LPO*HPO)))+
+CV*0.5)
= END.IF()

= ELSE.IF(NOM<>NOB)

=   SET.NAME("DRAGF",2*DMP*MMV*(NOM-NOB)/NOM)
=   IF((WAVDF+WINF+CUF)<DRAGF)
=     SET.NAME("VM",0)
=     RETURN()
=   END.IF()

=   IF((WINF+WAVDF)>DRAGF)
=     IF(CUD=1)
=       SET.NAME("VM",SQRT((WINF+WAVDF-DRAGF)/(PARACUR*2.85*((4*DCC+DCE)-
*(DR+SUR)*0.7+2*LPO*HPO)))+CV)
=     ELSE.IF(CUD=2)
=       SET.NAME("VM",SQRT((WINF+WAVDF-DRAGF)/(PARACUR*2.85*((4*DCC+DCE)-
*(DR+SUR)*0.7+2*LPO*HPO)))+CV*0.75)
=     ELSE.IF(CUD=3)
=       SET.NAME("VM",SQRT((WINF+WAVDF-DRAGF)/(PARACUR*2.85*((4*DCC+DCE)-
*(DR+SUR)*0.7+2*LPO*HPO)))+CV*0.5)
=     END.IF()
=   ELSE.IF((WINF+WAVDF)<DRAGF)
=     IF(CUD=1)
=       SET.NAME("VM",CV-SQRT((DRAGF-WINF-WAVDF)/(PARACUR*2.85*((4*DCC+DCE)-
*(DR+SUR)*0.7+2*LPO*HPO))))
=     ELSE.IF(CUD=2)
=       SET.NAME("VM",CV*0.75-SQRT((DRAGF-WINF-WAVDF)/(PARACUR*2.85*((4*DCC+DCE)-
*(DR+SUR)*0.7+2*LPO*HPO))))
=     ELSE.IF(CUD=3)
=       SET.NAME("VM",CV*0.5-SQRT((DRAGF-WINF-WAVDF)/(PARACUR*2.85*((4*DCC+DCE)-
*(DR+SUR)*0.7+2*LPO*HPO))))
=     END.IF()
=   END.IF()

= END.IF()

=END.IF()

=IF(VM<0)
= ALERT("VM < 0",3)

```

```
= HALTO  
=END.IF()  
  
=RETURN()
```

Check for Collision Happen

```
SUBCOLLISION()
```

```
=SET.NAME("XX1",XM)  
=SET.NAME("YY1",YM)  
=SET.NAME("XX2",XM+VM*DT*COS(WD))  
=SET.NAME("YY2",YM+VM*DT*SIN(WD))  
=ACTIVATE("[MODU.XLW]MODU.XLS")  
=EXTRACT(TRUE)  
  
=IF('B:\[MODU.XLW]MODU.XLS'!end<>"SAFE")  
  
= SET.NAME("pc",'B:\[MODU.XLW]MODU.XLS'!Ppc)  
= SET.NAME("COLLISIONa",COLLISIONa+1)  
= SET.NAME("COLLISION",COLLISION+pc)  
= SET.NAME("COLL",1)  
  
= IF(AND(Spe,'B:\[MODU.XLW]MODU.XLS'!spec="TAR"))  
=   SET.NAME("spnum",'B:\[MODU.XLW]MODU.XLS'!Tarnum)  
=   SPECTAR()  
= END.IF()  
  
= IF(AND(REPORT=1,ST=2))  
=   DIALOG.BOX(COLLHAPP)  
=   IF(CONT<>1)  
=     SET.NAME("stop",1)  
=   END.IF()  
= END.IF()  
  
=ELSE.IF('B:\[MODU.XLW]MODU.XLS'!end="SAFE")  
=   SET.NAME("COLL",0)  
=ELSE()  
=   ALERT("DATABASE ERROR",3)  
=   HALTO  
=END.IF()  
  
=RETURN()
```

Record Spectral Target Collision Number

```
SPECTARO
=IF(spnum=1)
= SET.NAME("Tarcoll1",Tarcoll1+pc)
=ELSE.IF(spnum=2)
= SET.NAME("Tarcoll2",Tarcoll2+pc)
=ELSE.IF(spnum=3)
= SET.NAME("Tarcoll3",Tarcoll3+pc)
=ELSE.IF(spnum=4)
= SET.NAME("Tarcoll4",Tarcoll4+pc)
=ELSE.IF(spnum=5)
= SET.NAME("Tarcoll5",Tarcoll5+pc)
=END.IF()
=RETURN()
```

Functions of Force calculation

```
FUNCTA
=RESULT(1)
=ARGUMENT("XMF",1)
=ARGUMENT("YMF",1)
=ARGUMENT("XHF",1)
=ARGUMENT("YHF",1)
=ARGUMENT("FIF",1)
=IF(ABS(XMF-XHF)<0.001)
= IF(YHF>YMF)
= SET.NAME("CTAF",2*PI()-FIF)
= ELSE()
= SET.NAME("CTAF",PI()-FIF)
= END.IF()
=ELSE()
= SET.NAME("KF",ATAN((YHF-YMF)/(XHF-XMF)))
= IF(XHF>XMF)
= SET.NAME("CTAF",1.5*PI()+KF-FIF)
= ELSE()
= SET.NAME("CTAF",PI()/2+KF-FIF)
= END.IF()
=END.IF()
=RETURN(CTAF)
```

FUNNDR

```
=RESULT(1)
=ARGUMENT("XMF",1)
=ARGUMENT("YMF",1)
=ARGUMENT("XHF",1)
=ARGUMENT("YHF",1)
```

```
=ARGUMENT("RMF",1)
=SET.NAME("NDRF",SQRT((XMF-XHF)^2+(YMF-YHF)^2)/RMF)
=RETURN(NDRF)
```

```
FUNWINSPEE
=RESULT(1)
=ARGUMENT("DPF",1)
=ARGUMENT("RMF",1)
=ARGUMENT("VFF",1)
=ARGUMENT("CTAF",1)
=ARGUMENT("NDRF",1)
=SET.NAME("WMF",0.885*(5.6*SQRT(DPF)-0.125*RMF)+VFF*COS(CTAF))
=IF(NDRF>1)
= SET.NAME("WF",1.944*WMF*NDRF ^(-0.38+0.08*COS(CTAF)))
=ELSE()
= SET.NAME("WF",1.944*1.047*WMF*(1-EXP(-3.1*NDRF)))
=END.IF()
=RETURN(WF)
```

```
FUNWINDIR
=RESULT(1)
=ARGUMENT("CTAF",1)
=SET.NAME("WDF",CTAF+(22+10*COS(CTAF))*PI()/180+PI()/2)
=RETURN(WDF)
```

```
FUNCURVEL
=RESULT(1)
=ARGUMENT("WF",1)
=ARGUMENT("WDMF",1)
=SET.NAME("SECF",2.5-1.5/XWD*WDMF)
=SET.NAME("CVF",0.025*WF*SECF)
=RETURN(CVF)
```

```
FUNWAVHEI
=RESULT(1)
=ARGUMENT("WF",1)
=ARGUMENT("WDMF",1)
=SET.NAME("SEWF",-0.4*(WDMF/XWD)^2+0.9*(WDMF/XWD)+0.5)
=SET.NAME("HMF",1.73*0.25*3.3*WF/1.944*SEWF)
=IF(HMF>0.6*WDMF)
= SET.NAME("HMF",0.6*WDMF)
=END.IF()
=RETURN(HMF)
```

FUNPERIOD	=RESULT(1) =ARGUMENT("WF",1) =ARGUMENT("CTAF",1) =SET.NAME("A1F",8-3.5*COS(CTAF)+2.7*SIN(CTAF)) =SET.NAME("A2F",0.143+0.138*COS(CTAF)-0.074*SIN(CTAF)) =SET.NAME("TF",A1F*WF^A2F) =RETURN(TF)
FUNSURGE	=RESULT(1) =ARGUMENT("HMF",1) =ARGUMENT("WDMF",1) =SET.NAME("SURF",0.03*HMF*(1+3*(XWD-WDMF)/XWD)) =RETURN(SURF)
FUNCURFOR	=RESULT(1) =ARGUMENT("PARACURF",1) =ARGUMENT("DRF",1) =ARGUMENT("SURF",1) =ARGUMENT("CVF",1) =ARGUMENT("DCCF",1) =ARGUMENT("DCEF",1) =ARGUMENT("LPOF",1) =ARGUMENT("HPOF",1) =ARGUMENT("CUDF",1) =IF(CUDF=1) = SET.NAME("CUFF",PARACURF*2.85*((4*DCCF+DCEF)*(DRF+SURF)*0.7+2*LPOF*HPOF)*CVF^2) =ELSE.IF(CUDF=2) = SET.NAME("CUFF",PARACURF*2.85*((4*DCCF+DCEF)*(DRF+SURF)*0.7*0.75^2 +2*LPOF*HPOF*0.5^2)*CVF^2) =ELSE.IF(CUDF=3) = SET.NAME("CUFF",PARACURF*2.85*((4*DCCF+DCEF)*(DRF+SURF)*0.7*0.5^2)*CVF^2) =ELSE() =HALTO =END.IF() =RETURN(CUFF)
FUNWAVFOR	=RESULT(1) =ARGUMENT("PARAWAVF",1) =ARGUMENT("DRF",1) =ARGUMENT("SURF",1)

```
=ARGUMENT("HMF",1)
=ARGUMENT("TF",1)
=ARGUMENT("WDMF",1)
=ARGUMENT("DCCF",1)
=ARGUMENT("DCEF",1)
=ARGUMENT("LPOF",1)
=ARGUMENT("HPOF",1)
=ARGUMENT("WPOF",1)
=SET.NAME("LF",5.12*TF^2*SQRT(TANH(2*PI()*WDMF/(5.12*TF^2))))
=SET.NAME("WA1F",32*HMF*PI()/LF*0.25)
=IF((WDMF-DRF)>0)
=SET.NAME("WA2F",WA1F*COSH(2*PI()*(WDMF-DRF)/LF)/COSH(2*PI()*WDMF/LF))
=ELSE()
=SET.NAME("WA2F",WA1F/COSH(2*PI()*WDMF/LF))
=END.IF()
=SET.NAME("WAaF",WA1F+WA2F/2)
=SET.NAME("WAF1F",PARAWAVF*2*PI()/4*(4*DCCF^2+DCEF^2)*(DRF+SURF+HMF/2)*WAaF)
=SET.NAME("WAF2F",PARAWAVF*2*LPOF*HPOF*WPOF*WA2F)
=SET.NAME("WAFF",WAF1F+WAF2F)
=RETURN(WAFF)
```

FUNWINFOR

```
=RESULT(1)
=ARGUMENT("PARAWINF",1)
=ARGUMENT("WF",1)
=ARGUMENT("EWAF",1)
=SET.NAME("WINFF",PARAWINF*0.0034*EWAF*WF^2)
=RETURN(WINFF)
```

FUNWAVDRI

```
=RESULT(1)
=ARGUMENT("HMF",1)
=IF(HMF>2)
=SET.NAME("WAVDFF",45000/28*HMF/1.73-5000/4)
=ELSE()
=SET.NAME("WAVDFF",0)
=END.IF()
=RETURN(WAVDFF)
```

FUNXH

```
=RESULT(1)
=ARGUMENT("XMF",1)
=ARGUMENT("YMF",1)
=ARGUMENT("FIF",1)
=ARGUMENT("XOF",1)
=ARGUMENT("RMF",1)
```

```

=SET.NAME("D1F",ABS(TAN(FIF)*XMF-YMF-XOF*TAN(FIF))/SQRT(TAN(FIF)^2+1))
=SET.NAME("D2F",SQRT(XMF^2+(YMF+XOF*TAN(FIF))^2))

=IF(D1F>(10*RMF))
=SET.NAME("XHF",1000)
=RETURN(XHF)
=END.IF()

=IF(FIF<(PI()/2))
=SET.NAME("XHF",SQRT((10*RMF)^2-D1F^2)-SQRT(D2F^2-D1F^2)*(-COS(FIF)))
=ELSE()
=SET.NAME("XHF",SQRT((10*RMF)^2-D1F^2)+SQRT(D2F^2-D1F^2)*(-COS(FIF)))
=END.IF()

=RETURN(XHF)

```

```

FUNWDM
=RESULT(1)
=ARGUMENT("WDMIF",1)
=ARGUMENT("YMIF",1)
=ARGUMENT("YMF",1)
=ARGUMENT("DTLF",1)
=SET.NAME("WDMF",DTLF-YMF)*WDMIF/(DTLF-YMIF)
=RETURN(WDMF)

```

```

SUBWRITE()
=ACTIVATE("[MODU.XLW]MODU.XLS")
=SELECT($C$1)
=SELECT.END(4)
=SELECT(OFFSET(SELECTION(),1,0))
=FORMULA("=[MODU.XLW]MODUSIM.XLMIXM",SELECTION())
=SELECT(OFFSET(SELECTION(),0,1))
=FORMULA("=[MODU.XLW]MODUSIM.XLMIYM",SELECTION())
=SELECT(OFFSET(SELECTION(),0,1))
=FORMULA("=[MODU.XLW]MODUSIM.XLMIXH",SELECTION())
=SELECT(OFFSET(SELECTION(),0,1))
=FORMULA("=[MODU.XLW]MODUSIM.XLMIYH",SELECTION())
=RETURN()

```

Calculate Probability of Collision within the Target Circle

```

LARGEP()

=SET.NAME("cc",0)
=FOR("I",1,200)
=SET.NAME("xox",RiskUniform(-RF,RF))
=SET.NAME("FIH",RiskUniform(0,180)*PI()/180)

```

```

=FOR("II",1,NM)
=SET.NAME("xf",RiskUniform(-RF,RF))
=SET.NAME("yf",RiskUniform(-RF,RF))

=IF(xf^2+yf^2-RF^2>0)
=  GOTO(PP)
=END.IF()

=IF((FIH-PI()<0.001)
=  SET.NAME("dd",ABS(xox-xf))
=ELSE.IF(OR(FIH=0,(FIH-PI())<0.001))
=  SET.NAME("dd",ABS(xox-yf))
=ELSE()
=  SET.NAME("dd",ABS(TAN(FIH)*xf-yf-TAN(FIH)*xox)/SQRT(TAN(FIH^2)+1)))
=END.IF()

=IF(dd<RS)
=  SET.NAME("cc",cc+1)
=  GOTO(PPP)
=END.IF()

=NEXT()
=NEXT()
=FORMULA("=CC/200",MODUSIM.XLMIJ645)
=ALERT("Simulation Finished!",3)
=DIALOG.BOX(LPRESU)
=RETURN()

```

Input Target Circle Information

FINFO()

=DIALOG.BOX(INFO)

=RETURN()

